

In the claims:

All of the claims standing for examination are reproduced below with appropriate status indication.

1-17. (Canceled)

18. (Currently amended) A method for receiving notification of ~~random~~ structural changes applied to electronic information pages accessed by a proxy network navigation and interaction system and effecting updates to navigation templates based on the change information, comprising steps of:

(a) establishing notification of ~~a failed~~ execution of one of the navigation templates navigation and interaction routine executed for the purpose of navigating to and interacting with ~~an~~ electronic information ~~page~~ pages on a data-packet-network;

(b) recording ~~an instance of the failed routine including~~ parameters associated with the cause of failure, including identification of at least one ~~of a plurality of~~ modular logic ~~blocks~~ block ~~used to build each of the navigation templates~~ involved at the point of failure;

(c) accessing the recorded instance of the failed ~~routine~~ navigation template for review purposes;

(d) navigating to the electronic information page identified in the recorded instance ~~on the data packet network;~~

(e) ~~accessing source~~ determining information ~~associated with electronic information page identified in the recorded instance~~ necessary to repair the logic block involved at the point of failure;

(f) creating a new modular logic block according to the ~~source~~ information ~~and according to information contained in the recorded instance;~~ and

(g) automatically installing the newly created modular logic block into the navigation template that failed, and into all existing navigation templates that depend on the updated information for successful function depend on the failed logic block.

19. (Original) The method of claim 18, wherein the data-packet-network is an Internet network and electronic information page is a web page hosted on the network.

20. (Previously presented) The method of claim 19, wherein the navigation template is a test navigation template created for the purpose.

21. (Previously presented) The method of claim 19, wherein the navigation template is a client navigation template executed to perform services for the client.

22. (Original) The method of claim 19 wherein in step (b), the recorded instance of a failed routine is created in the form of a data file and stored in a data repository accessible through the network.

23. (Original) The method of claim 22 wherein in step (c), the recorded instance of the failed navigation routine is accessed by a human software developer.

24. (Original) The method of claim 23 wherein in step (d), navigation is performed by the developer utilizing an instance of browser software installed on a computerized workstation.

25. (Previously presented) The method of claim 24 wherein in step (f), the modular logic block is installable to navigation templates.

26. (Previously presented) The method of claim 25 wherein in step (g), the modular logic block self-installs to a depended navigation template.

27. (Previously presented) The method of claim 18 wherein a step is added between steps (f) and (g) for testing the modular logic block before implementation.

28. (Previously presented) The method of claim 26 wherein in step (g) more than one modular logic block is created for a single navigation template.

29-86. (Canceled)